



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

B. Tech – IV Year I Semester

S.No.	Category	Title	L	T	P	C
1	Professional Core	Big Data Analytics	3	0	0	3
2	Management Course- II	Human Resource & Project Management	2	0	0	2
3	Professional Elective-IV	1. Software Architecture & Design Patterns 2. Block chain Technology 3. DevOps 4. Natural Language Processing 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
4	Professional Elective-V	1. Agile Methodologies 2. Expert Systems 3. Reinforcement Learning 4. High Performance Computing 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
5	Open Elective-III	1. Object oriented Programming Through Java	3	0	0	3
6	Open Elective-IV	1. Computer Networks 2. Software Engineering 3. IOT Based smart Systems 4. Quantum Science and Technology	3	0	0	3
7	Skill Enhancement Course	Full Stack Development -2 SWAYAM Plus - Certificate Course on Data Analytics	0	1	2	2
8	Audit Course	Constitution of India	2	0	0	-
9	Evaluation of Industry Internship / Mini Project		-	-	-	2
Total			19	01	02	21
MC	Student may select from the minors pool		3	0	0	3
HC	Student may select from the honors pool		3	0	0	3
HC	Student may select from the honors pool		3	0	0	3



B. Tech– IV Year II Semester

S.No.	Category	Title	L	T	P	Credits
1	Internship / Project Work	Full semester Internship / Project Work	0	0	24	12

Note : Student need to do at least ONE MOOC Course (3 credits out of 160 credits) to meet the mandatory requirement (11th criteria, as per R23 Regulations)

Open Electives, offered to other department students:

Open Elective I : Operating Systems / Computer Organization and

Architecture Open Elective II : Database Management Systems

Open Elective III : Object Oriented Programming Through Java

Open Elective IV : Computer Networks / Software Engineering / IOT Based Smart Systems
/Quantum Science and Technology

Minor Engineering

Note:

- 1. To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.*
- 2. During Minor/Honors Course selection, there should not be any overlapping with Regular/Major/OPEN Electives*

Minor in Data Science

1. Introduction to Data Science 3-0-3-4.5 (II-II)
2. Introduction to Database Systems 3-0-0-3 (III-I)
3. Data Engineering 3-0-3-4.5 (III-II)
4. ~~Deep~~ Machine Learning 3-0-0-3 (IV-I)

Any of the following 12 Week 3 credit NPTEL MOOC Courses

5. Operating Systems
6. Artificial Intelligence: Knowledge Representation and Reasoning
7. Computer Networks and Internet Protocol
8. Fundamentals of Object Oriented Programming
9. Discrete Mathematics for CS
10. Software Engineering
11. Deep Learning & Machine Learning

Honors:

Agentic AI

Quantum Machine Learning

Real Time Data Processing



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

Secure Computation (NPTEL)

Privacy Preserving Machine Learning*(NPTEL)



IV Year I Semester	BIG DATA ANALYTICS- V231414431 (Professional Core)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of the course are to

- Optimize business decisions and create competitive advantage with Big Data analytics
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- To introduce programming tools PIG & HIVE in Hadoop ecosystem.

COURSE OUTCOMES: Upon completion of this course, students will be able to

- CO1. Explain the basics of Big Data, Hadoop architecture, and its components
- CO2. Implement Java data structures and generics to develop Map Reduce programs
- CO3. Use Hadoop tools like Pig and Hive to process and analyze unstructured data
- CO4. Configure and deploy Hadoop clusters, and develop Map Reduce programs to solve business Problems
- CO5. Apply Spark concepts and Stream processing techniques to handle large datasets

UNIT-I: Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

UNIT-II: Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) Building blocks of Hadoop (Namenode, Data node, Secondary Namenode, Job Tracker, and Task Tracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III: Writing Map Reduce Programs: A Weather Dataset, Understanding Hadoop API for Map Reduce Framework (Old and New), Basic programs of Hadoop Map Reduce: Driver code, Mapper code, Reducer code, Record Reader, Combiner, Practitioner

UNIT-IV: Stream Memory and Spark: Introduction to Streams Concepts– Stream Data Model and Architecture , Stream computing, Sampling Data in a Stream , Filtering Streams ,Counting Distinct Elements in a Stream , Introduction to Spark Concept , Spark Architecture and components , Spark installation , Spark RDD(Resilient Distributed Dataset) – Spark RDD operations.

UNIT-V: Pig: Hadoop Programming Made Easier Admiring the Pig Architecture, Going



with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin.

Applying Structure to Hadoop Data with Hive: Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing data

TEXT BOOKS:

1. Wiley & Big Java 4th Edition, Cay Horstmann, Wiley John Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly

REFERENCE BOOKS:

1. Hadoop in Action by Chuck Lam, MANNING Publ.
2. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk, Bruce Brown, Rafael Coss
3. Hadoop in Practice by Alex Holmes, MANNING Publ.
4. Big Data Analytics by Dr. A.Krishna Mohan and Dr.E.Laxmi Lydia
5. Hadoop Map Reduce Cookbook, SrinathPerera, ThilinaGunarathne

Software Links:

1. [Hadoop:http://hadoop.apache.org/](http://hadoop.apache.org/)
2. Hive: <https://cwiki.apache.org/confluence/display/Hive/Home>
3. Piglatin: <http://pig.apache.org/docs/r0.7.0/tutorial.html>



IV Year I Semester	HUMAN RESOURCES AND PROJECT MANAGEMENT-V2314144M1 (Management Course-II)	L	T	P	C
		2	0	0	2

COURSE OBJECTIVES: The main objectives of the course are to

- Provide knowledge about HR planning, recruitment, selection, and job design.
- Develop skills in managing HR functions such as performance appraisal, compensation, and employee relations.
- Emphasize the importance of ethical practices and HR audits in maintaining organizational health.
- Understand the HRD framework and its impact on organizational success.
- Improve group interaction and team dynamics for better collaboration and performance.
- Implement appropriate management strategies tailored to specific challenges in different project types.

COURSE OUTCOMES: Upon completion of this course, students will be able to

CO 1. Explain the nature, scope, and functions of HRM, and emerging trends in HRM

CO 2. Implement performance appraisal, compensation, and employee relations strategies to improve organizational performance

CO 3. Apply group interaction and team dynamics principles to enhance collaboration and performance

CO 4. Explain project management concepts, project networks, and project life cycle

CO 5. Implement project management strategies tailored to specific project types and challenges

UNIT –I: HRM: Nature, Scope, Concept of HRM, Functions of HRM, Role of HR manager, emerging trends in HRM, E-HRM, HR audit models, ethical aspects of HRM. HR Planning, Demand and Supply forecasting of HR, Job Design, Recruitment, Sources of recruitment, Selection- Selection Procedure.

UNIT –II: HRD, HR accounting, Models, Concept of Training and Development, Methods of Training. Performance Appraisal: Importance Methods of performance appraisal, Career Development and Counseling, group interaction.

UNIT –III: Basics of Project Management, Concept, resource management, Project environment, Types of Projects, project networks-DPR, Project life cycle, Project proposals, Monitoring project progress, Project appraisal and Project selection, 80-20 rules, production technology, communication matrix

UNIT-IV: Identify various project types and their unique management challenges and



apply appropriate management strategies for each. Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

UNIT-V: Project Implementation and Review: Forms of project organization, project planning, project control, human aspects of project management, prerequisites for successful project implementation, project review, performance evaluation, abandonment analysis

Text Books:

1. Robert L. Mathis, John H. Jackson, Manas Ranjan Tripathy, Human Resource Management, Cengage Learning 2016.
2. Sharon Pande and Swapnalekha Basak, Human Resource Management, Text and Cases, Vikas Publishing, 2e, 2016.
3. Stewart R. Clegg, Torgeir Skyttermoen, Anne Live Vaagaasar, Project Management, Sage Publications, 1e, 2021.
4. K. Nagarajan, Project Management, New Age International Publishers, 8e, 2017.

Reference Books:

1. Subba Rao P, "Personnel and Human Resource Management-Text and Cases", Himalaya Publications, Mumbai, 2013.
2. K Aswathappa, "Human Resource and Personnel Management", Tata McGraw Hill, New Delhi, 2013.
3. Prasanna Chandra, "Projects, Planning, Analysis, Selection, Financing, Implementation and Review", Tata McGraw Hill Company Pvt. Ltd., New Delhi, 1998.
4. Vasanth Desai, "Project Management", 4th edition, Himalaya Publications, 2018.
5. Lalitha Balakrishnan, Gowri, "Project Management", Himalaya publishing house, New Delhi, 2022.



IV Year I Semester	SOFTWARE ARCHITECTURE & DESIGN PATTERNS-V231414441 (Professional Elective-IV)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of the course are to

- Understand the basic concepts to identify state behavior of real world objects
- Apply Object Oriented Analysis and Design concepts to solve complex problems
- Construct various UML models using the appropriate notation for specific problem context
- Design models to Show the importance of systems analysis and design in solving complex problems using case studies
- Study of Pattern Oriented approach for real world problems

COURSE OUTCOMES: Upon completion of this course, students will be able to

- CO1.Explain the key concepts of object-oriented design, benefits, and drawbacks of the paradigm
- CO2. Use object-oriented analysis and design concepts to solve complex problems
- CO3. Apply design patterns to solve design problems and create effective system designs
- CO4. Analyze complex problems, design systems, and implement solutions using object-oriented principles.
- CO5. Explain the concepts of distributed objects, client-server systems, and web services

UNIT – I: Introduction: What is a design pattern? Describing design patterns, the catalog of design pattern, organizing the catalog, how design patterns solve design problems, how to select a design pattern, how to use a design pattern What is object oriented development? key concepts of object oriented design other related concepts, benefits and drawbacks of the paradigm

UNIT – II: Analysis a System: Overview of the analysis phase, stage 1 gathering the requirements functional requirements specification, defining conceptual classes and relationships, using the knowledge of the domain Design and Implementation, discussions and further reading

UNIT – III: Design Pattern Catalog: Structural patterns, Adapter, bridge, composite, decorator, facade, flyweight, and proxy.

UNIT – IV: Interactive systems and the MVC architecture: Introduction The MVC architectural pattern, analyzing a simple drawing program designing the system, designing of the subsystems, getting into implementation, implementing undo operation drawing incomplete items, adding a new feature pattern based solutions

UNIT – V: Designing with Distributed Objects: Client server system, java remote method



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

invocation, implementing an object oriented system on the web, Web services (SOAP, Restful), Enterprise Service Bus

Text Books:

1. Object oriented analysis, design and implementation, Brahma dathan, sarnath rammath , universities press,2013
2. Design patterns, Erich Gamma, Richard helan , Ralph johman , john vlissides, PEARSON Publication,2013

Reference Books:

1. Frank Bachmann, Regine Meunier, Hans Rohnert “Pattern Oriented Software Architecture” Volume 1, 1996.
2. William J Brown et al., "Anti Patterns: Refactoring Software, Architectures and Projects in Crisis", John Wiley, 1998



IV Year I Semester	BLOCK CHAIN TECHNOLOGY-V231414442 (Professional Elective-IV)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to

- Discuss the fundamental cryptographic concepts, such as hashing, encryption, and digital signatures, used in Block chain technology
- Explore the technologies borrowed from other fields, such as hash pointers and digital cash, and their application in Block chain
- Describe various Block chain models, including permission and permission less platforms, and their characteristics
- Discuss the Ethereum platform, its Virtual Machine (EVM), and the concept of Smart Contracts
- Introduce Hyper ledger Fabric, a permissioned Blockchain platform, and its applications in various industries

COURSE OUTCOMES: At the end of the Course student will be able to

- CO1. Discuss the Cryptographic primitives used in Block chain
- CO2. Discuss about various technologies borrowed in Block chain
- CO3. Illustrate various models for block chain
- CO4. Discuss about Ethereum
- CO5. Discuss about Hyper ledger Fabric

UNIT I: INTRODUCTION TO BLOCKCHAIN

Introduction, history of Bit coin and origins of Block chain, Fundamentals of Block chain and key components (**Chapter 1-book1**), Permission and Permission-less platforms (**Chapter 1-book2**), Introduction to Cryptography, SHA256 and ECDSA, Hashing and Encryption, Symmetric/ Asymmetric keys, Private and Public Keys(**Chapter 3-book2**).

UNIT II: TECHNOLOGIES BORROWED IN BLOCKCHAIN

Technologies Borrowed in Block chain –hash pointers- - Digital cash etc.- Bitcoin blockchain - Wallet – Blocks Merkley Tree - hardness of mining - Transaction verifiability - Anonymity - forks
- Double spending - Mathematical analysis of properties of Bitcoin - Bitcoin- the challenges and solutions. (**Chapter 3-book2**).

UNIT III: CONSENSUS MECHANISMS

Consensus Algorithms: Proof of Work (PoW) as random oracle - Formal treatment of consistency- Liveness and Fairness - Proof of Stake (PoS) based Chains -Hybrid models (PoW + PoS), Byzantine Models of fault tolerance. ((**Chapter 1-book2**))



UNIT IV: ETHEREUM

Ethereum -Ethereum Virtual Machine (EVM) -Wallets for Ethereum -Solidity - Smart Contracts (**Chapter 5-book1**), - The Turing Completeness of Smart Contract Languages and verification challenges- Using smart contracts to enforce legal contracts- Comparing Bitcoin scripting vs. Ethereum Smart Contracts-Some attacks on smart contracts (**Chapter 6 and Chapter 7-book2**)

UNIT V: HYPERLEDGER FABRIC

Hyper ledger fabric- the plug and play platform and mechanisms in permissioned block chain

- Beyond Crypto currency – applications of blockchain in cyber security- integrity of information- E-Governance and other contract enforcement mechanisms - Limitations of blockchain as a technology and myths vs reality of blockchain technology (**Chapter 16-book1**), (**Chapter 9-book2**)

TEXT BOOKS:

1. Block chain Technology Chandramouli Subramanian, Asha A George, Abhilash K A and Meena Karthikeyan, University Press, 2020.
2. Mastering Blockchain - Distributed ledger technology, decentralization, and smart contracts explained, Imran Bashir, 2nd ed. Edition, 2018, packet publication

REFERENCES:

1. S.Shukla, M.Dhawan, S.Sharma, S.Venkatesan “Block chain Technology: Crypto currency and Applications” , Oxford University Press 2019 .
2. Cryptography and network security principles and practice, William Stallings, Pearson, 8th Edition,

WEB REFERENCES:

1. <https://drive.google.com/file/d/1PtYaDmWYaqPVGjKDnMYGWO5eoI5wMPtJ/view>
2. <https://archive.nptel.ac.in/courses/106/104/106104220/>
3. <https://www.tutorialspoint.com/blockchain/index.htm>



IV Year I Semester	DEVOPS-V231414443 (Professional Elective-IV)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to

- Describe the agile relationship between development and IT operations.
- Understand the skill sets and high-functioning teams involved in DevOps and related methods to reach a continuous delivery capability.
- Implement automated system updates and the DevOps lifecycle.

COURSE OUTCOMES: On successful completion of this course, students will be able to

- CO1. Identify the DevOps Concepts and Tools for effective project management.
- CO2. Utilize GIT to keep track of different versions of the source code.
- CO3. Build and Automate Tests using Jenkins.
- CO4. Implement containerization with Docker.
- CO5. Use ANSIBLE, Kubernetes for automation and deployment.

UNIT-I: Introduction to DevOps: Introduction to SDLC, Agile Model. Introduction to Devops. DevOps Features, DevOps Architecture, DevOps Lifecycle, Understanding Workflow and principles, Introduction to DevOps tools, Build Automation, Delivery Automation, Understanding Code Quality, Automation of CI/ CD. Release management, Scrum, Kanban, delivery pipeline, bottlenecks, and examples

UNIT-II: Source Code Management (GIT): The need for source code control, the history of source code management, Roles and code, source code management system and migrations. What is Version Control and GIT, GIT Installation, GIT features, GIT workflow, working with remote repository, GIT commands, GIT branching, GIT staging and collaboration. **UNIT TESTING- CODE COVERAGE:** Junit, n Unit & Code Coverage with Sonar Qube, SonarQube - Code Quality Analysis.

UNIT-III: Build Automation - Continuous Integration (CI): Build Automation, What is CI Why CI is Required, CI tools, Introduction to Jenkins (With Architecture), Jenkins workflow, Jenkins master slave architecture, Jenkins Pipelines, PIPELINE BASICS - Jenkins Master, Node, Agent, and Executor Freestyle Projects& Pipelines, Jenkins for Continuous Integration, Create and Manage Builds, User Management in Jenkins Schedule Builds, Launch Builds on Slave Nodes.

UNIT-IV: Continuous Delivery: Importance of Continuous Delivery, CONTINUOUS DEPLOYMENT CD Flow, Containerization with Docker: Introduction to Docker, Docker installation, Docker commands, Images & Containers, Docker File, Running containers, working with containers and publish to Docker Hub. Testing Tools: Introduction to Selenium and its features, Java Script testing.

UNIT-V: Configuration Management - ANSIBLE: Introduction to Ansible, Ansible tasks



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

Roles, Jinja2 templating, Vaults, Deployments using Ansible. CONTAINERIZATION
USING KUBERNETES (OPENSIFT): Introduction to Kubernetes
Namespace & Resources, CI/CD

- On OCP, BC, DC & Config Maps, Deploying Apps on Open shift Container Pods.
Introduction



to Puppet master and Chef.

TEXT BOOKS

1. Joyner, Joseph., Devops for Beginners: Devops Software Development Method Guide for Software Developers and It Professionals, 1st Edition Mihails Konoplows, 2015.
2. Allison Machado de Menezes., Hands-on DevOps with Linux, 1st Edition, BPB Publications, India, 2021.

REFERENCE BOOKS

1. Len Bass, Ingo Weber, Liming Zhu. DevOps: A Software Architect's Perspective. Addison Wesley; ISBN-10
2. Gene Kim Je Humble, Patrick Debois, John Willis. The DevOps



IV Year I Semester	NATURAL LANGUAGE PROCESSING V231414444 (Professional Elective -IV)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: This course introduces the fundamental concepts and techniques of natural language processing (NLP).

- Students will gain an in-depth understanding of the computational properties of natural languages and the commonly used algorithms for processing linguistic information.
- The course examines NLP models and algorithms using both the traditional symbolic and the more recent statistical approaches.
- Enable students to be capable to describe the application based on natural language processing and to show the points of syntactic, semantic and pragmatic processing.

Course Outcomes: After completion of this course, students will be able to

- CO1. Demonstrate a given text with basic Language features
- CO2. Design an innovative application using NLP components
- CO3. Explain a rule based system to tackle morphology/syntax of a language
- CO4. Design a tag set to be used for statistical processing for real-time applications
- CO5. Compare and contrast the use of different statistical approaches for different types Of NLP applications.

UNIT I: INTRODUCTION: Origins and challenges of NLP – Language Modeling: Grammar-based LM, Statistical LM – Regular Expressions, Finite-State Automata – English Morphology, Transducers for lexicon and rules, Tokenization, Detecting and Correcting Spelling Errors, Minimum Edit Distance.

UNIT II: WORD LEVEL ANALYSIS: Unsmoothed N-grams, Evaluating N-grams, Smoothing, Interpolation and Back off – Word Classes, Part- of-Speech Tagging, Rule-based, Stochastic and Transformation-based tagging, Issues in PoS tagging – Hidden Markov and Maximum

UNIT III: SYNTACTIC ANALYSIS: Context-Free Grammars, Grammar rules for English, Treebanks, Normal Forms for grammar – Dependency Grammar – Syntactic Parsing, Ambiguity, Dynamic Programming parsing – Shallow parsing Probabilistic CFG, Probabilistic CYK, Probabilistic Lexicalized CFGs – Feature structures, Unification of feature structures

UNIT IV: SEMANTICS AND PRAGMATICS: Requirements for representation, First-Order Logic, Description Logics – Syntax-Driven Semantic analysis, Semantic attachments – Word Senses, Relations between Senses, Thematic Roles, selectional restrictions – Word Sense Disambiguation, WSD using Supervised, Dictionary & Thesaurus, Bootstrapping



methods – Word Similarity using Thesaurus and Distributional methods.

UNIT V: DISCOURSE ANALYSIS AND LEXICAL RESOURCES: Discourse segmentation, Coherence – Reference Phenomena, Anaphora Resolution using Hobbs and Centering Algorithm – Co reference Resolution – Resources: Porter Stemmer, Lemmatizer, Penn Treebank, Brill's Tagger, Word Net, Prop Bank, Frame Net, Brown Corpus, British National Corpus (BNC), Tuning.



Text Books:

1. Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech, 2nd Edition, Daniel Jurafsky, James H. Martin - Pearson Publication, 2014.
2. Natural Language Processing with Python, First Edition, Steven Bird, Ewan Klein and Edward Loper, O'Reilly Media, 2009.

Reference Books:

1. Language Processing with Java and Ling Pipe Cookbook, 1st Edition, Breck Baldwin, Atlantic Publisher, 2015.
2. Natural Language Processing with Java, 2nd Edition, Richard M Reese, O'Reilly Media, 2015.
3. Handbook of Natural Language Processing, Second, Nitin Indurkha and Fred J. Damerau, Chapman and Hall/CRC Press, 2010. Edition
4. Natural Language Processing and Information Retrieval, 3rd Edition, Tanveer Siddiqui, U.S. Tiwary, Oxford University Press, 2008.



IV Year I Semester	AGILE METHODOLOGIES-V231414446 (Professional Elective-V)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to:

- Explain the principles, values, and practices of Agile methodology and its application in software development.
- Discuss the importance of knowledge sharing and management in Agile teams, and apply techniques such as Story-Card Maturity Model (SMM)
- Explain the impact of Agile processes on requirements engineering, and apply Agile requirements management and prioritization techniques
- Discuss Agile approaches to quality assurance, testing, and metrics, including Test-Driven Development (TDD) and Feature Driven Development (FDD).
- Examine the pros and cons of working in Agile teams, and discuss strategies for effective team collaboration and communication.

Course Outcomes: At the end of the course the student will be able to

CO1. Apply agile methodology and agile process to create high quality Software.

CO2. Use Agile methodology for knowledge management.

CO3. Apply Agile development and testing techniques to manage risks.

CO4. Analyze the pros and cons of working in Agile Team.

CO5. Apply Feature Driven Development on large size projects.

UNIT –I: Agile Methodology: Theories for Agile Management, Agile Software Development – Traditional Model vs. Agile Model, Classification of Agile Methods, Agile Manifesto and Principles, Agile Project Management, Agile Team Interactions, Ethics in Agile Teams, Agility in Design, Testing, Agile Documentations, Agile Drivers, Capabilities and Values.

UNIT-II: Agile Process: Lean Production - SCRUM, Crystal, Feature Driven Development- Adaptive Software Development - Extreme Programming: Method Overview – Lifecycle – Work Products, Roles and Practices.

UNIT-III: Agile Knowledge Sharing – Role of Story-Cards – Story-Card Maturity Model (SMM).

UNIT-IV: Agility and Requirements Engineering: Impact of Agile Processes in RE, Current Agile Practices, Variance, Overview of RE Using Agile, Managing Unstable Requirements, Requirements Elicitation, Agile Requirements Abstraction Model, Requirements Management in Agile Environment, Agile Requirements Prioritization, Agile Requirements Modeling, Generation – Concurrency in Agile Requirements Generation.

UNIT-V: Agility and Quality Assurance: Agile Product Development, Agile Metrics,



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

Feature Driven Development (FDD), Financial and Production Metrics in FDD, Agile Approach to Quality Assurance, Test Driven Development, Agile Approach in Global Software Development.

TEXT BOOKS:

1. David J. Anderson and Eli Schragenheim, Agile Management for Software Engineering: Applying the Theory of Constraints for Business Results, Pearson, 2003.
2. Hazza and Dubinsky, Agile Software Engineering, Springer, 2009th edition, 2008.

REFERENCE BOOKS:

1. Craig Larman, —Agile and Iterative Development, Addison-Wesley, First Edition, 2003.



IV Year I Semester	EXPERT SYSTEMS-V231414447 (Professional Elective-V)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to

- Explain the basics of AI programming languages and various search strategies, including blind and heuristic search techniques
- Describe knowledge representation issues, including predicate logic, semantic nets, frames, and inheritance, and constraint propagation
- Explain the basics of expert systems, their architecture, and the types of problems they can handle.
- Describe the tools and techniques used in expert system development, including knowledge representation, knowledge engineering, and system-building aids
- Explain the steps involved in building an expert system, including selecting the tool, acquiring knowledge, and dealing with difficulties and pitfalls.

Course Outcomes: At the end of the course, students will be able to

- CO1. Apply blind and heuristic search techniques to solve problems
- CO2. Represent knowledge using predicate logic, semantic nets, frames, and rules
- CO3. Design and develop expert systems using various tools and techniques
- CO4. Analyze the process of building an expert system, including knowledge acquisition and tool selection
- CO5. Evaluate the performance of expert systems and identify common pitfalls and difficulties

UNIT - I: Introduction to AI programming languages, Blind search strategies, Breadth-first – Depth- first –Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game tress, Minmax algorithms, game playing – Alpha-beta pruning.

UNIT - II: Knowledge representation issues predicate logic – logic programming Semantic nets- frames and inheritance, constraint propagation; Representing Knowledge using rules, Rules-based deduction systems.

UNIT - III: Introduction to Expert Systems, Architecture of expert systems, Representation and organization of knowledge, Basics characteristics, and types of problems handled by expert systems.

UNIT - IV: Expert System Tools: Techniques of knowledge representations in expert systems, knowledge engineering, system-building aids, support facilities, stages in the development of expert systems.

UNIT - V: Building an Expert System: Expert system development, Selection of the tool, Acquiring Knowledge, Building process. Problems with Expert Systems: Difficulties, common pitfalls in planning, dealing with domain experts, difficulties during development.



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

TEXT BOOKS:

1. Elain Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw-Hill, New Delhi.
2. Waterman D.A., “A Guide to Expert Systems”, Addison Wesley Longman.

REFERENCE BOOKS:

1. Stuart Russel and other Peter Norvig, “Artificial Intelligence – A Modern Approach”, Prentice Hall.
2. Patrick Henry Winston, “Artificial Intelligence”, Addison Wesley.
3. Patterson, Artificial Intelligence & Expert System, Prentice Hall India, 1999.
4. Hayes-Roth, Lenat, and Waterman: Building Expert Systems, Addison Wesley.
5. Weiss S.M. and Kulikowski C.A., “A Practical Guide to Designing Expert Systems”, Rowman & Allanheld, New Jersey



IV Year I Semester	REINFORCEMENT LEARNING-V231414448 (Professional Elective-V)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVE: The main objectives of this course are to

- Explain the basics of reinforcement learning, its elements, limitations, and scope.
- Discuss the n-armed bandit problem, action-value methods, and various techniques like incremental implementation, optimistic initial values, and upper-confidence-bound action selection
- Describe the agent-environment interface, goals, rewards, value functions, and optimal value functions in Markov decision processes
- Introduce Monte Carlo Methods: Explain Monte Carlo prediction, estimation of action values, control, and off-policy prediction via importance sampling.
- Discuss various applications and case studies of reinforcement learning, such as TD-Gammon, Acrobot, elevator dispatching, and job-shop scheduling.

Course Outcomes: At the end of the course, students will be able to

CO1. Enumerate the elements of Reinforcement Learning

CO2. Solve then-armed Bandit problem

CO3. Compare different Finite Mark Decision Process

CO4. Discuss about Monte Carlo Methods in solving real world problems

CO5. Describe the Applications and Case Studies of Reinforcement Learning

UNIT-I: The Reinforcement Learning Problem: Reinforcement Learning, Examples, Elements of Reinforcement Learning, Limitations and Scope, an Extended Example: Tic-Tac-Toe, Summary, History of Reinforcement Learning.

UNIT-II: Multi-arm Bandits: An n-Armed Bandit Problem, Action-Value Methods, Incremental Implementation, tracking a Non stationary Problem, Optimistic Initial Values, Upper-Confidence- Bound Action Selection, Gradient Bandits, Associative Search (Contextual Bandits)

UNIT-III: Finite Markov Decision Processes: The Agent–Environment Interface, Goals and Rewards, Returns, Unified Notation for Episodic and Continuing Tasks, The Markov Property, Markov Decision Processes, Value Functions, Optimal Value Functions, Optimality and Approximation.

UNIT-IV: Monte Carlo Methods: Monte Carlo Prediction, Monte Carlo Estimation of Action Values, Monte Carlo Control, Monte Carlo Control without Exploring Starts, Off-policy Prediction via Importance Sampling, Incremental Implementation, Off-Policy Monte Carlo Control, Importance Sampling on Truncated Returns

UNIT-V: Applications and Case Studies: TD-Gammon, Samuel’s Checkers Player, the



Acrobot, Elevator Dispatching, Dynamic Channel Allocation, Job-Shop Scheduling.

Text Books:

1. Richard S.Sutton and Andrew G.Barto, "Reinforcement Learning- An Introduction", 2nd Edition, The MIT Press, 2018
2. Marco Wiering, Martijn van Otterlo Reinforcement Learning: State-of-the-Art (Adaptation, Learning and Optimization(12)) 2012th Edition

Reference Books:

1. Vincent François-Lavet, Peter Henderson, Riashat Islam, An Introduction to Deep Reinforcement Learning (Foundations and Trends(r) in Machine Learning) , 2019.



IV Year I Semester	HIGH PERFORMANCE COMPUTING V231414449 (Professional Elective-V)	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to

- Explain the basics of parallel computing, its scope, and trends in microprocessor architectures, and limitations of memory systems
- Describe principles of parallel algorithm design, decomposition techniques, task mapping, and load balancing, and apply these concepts to solve problems
- Explain basic communication operations, such as broadcast, reduction, and scatter-gather, and improve their performance
- Use analytical models to evaluate the performance of parallel programs, identify sources of overhead, and optimize granularity
- Implement parallel algorithms for sorting, graph problems, and matrix operations, and use CUDA architecture and CUDA C for parallel programming

COURSE OUTCOMES: On completion of the course, student will be able to

- CO1. Describe different parallel architectures, inter-connect networks, programming models
CO2. Develop an efficient parallel algorithm to solve given problem
CO3. Analyze and measure performance of modern parallel computing systems
CO4. Build the logic to parallelize the programming task
CO5. Implement parallel algorithms for sorting, graph problems, and use CUDA architecture for parallel programming.

Unit I Introduction: Motivating Parallelism, Scope of Parallel Computing, Parallel Programming Platforms: Implicit Parallelism, Trends in Microprocessor and Architectures, Limitations of Memory, System Performance, Dichotomy of Parallel Computing Platforms, Physical Organization of Parallel Platforms, Communication Costs in Parallel Machines, Scalable design principles, Architectures: N-wide superscalar architectures, Multi-core architecture.

Unit II Parallel Programming: Principles of Parallel Algorithm Design: Preliminaries, Decomposition Techniques, Characteristics of Tasks and Interactions, Mapping Techniques for Load Balancing, Methods for Containing Interaction Overheads, Parallel Algorithm Models, The Age of Parallel Processing, the Rise of GPU Computing, A Brief History of GPUs, Early GPU.

Unit III Basic Communication: Operations- One-to-All Broadcast and All-to-One Reduction, All- to-All Broadcast and Reduction, All-Reduce and Prefix-Sum Operations, Scatter and Gather, All- to-All Personalized Communication, Circular Shift, Improving the Speed of Some Communication Operations. Programming shared address space platforms: threads- basics, synchronization, Open MP programming



Unit IV: Analytical Models: Sources of overhead in Parallel Programs, Performance Metrics for Parallel Systems, and The effect of Granularity on Performance, Scalability of Parallel Systems, Minimum execution time and minimum cost, optimal execution time. Dense Matrix Algorithms: Matrix Vector Multiplication, Matrix-Matrix Multiplication.

Unit V: Parallel Algorithms- Sorting and Graph : Issues in Sorting on Parallel Computers, Bubble Sort and its Variants, Parallelizing Quick sort, All-Pairs Shortest Paths, Algorithm for sparse graph, Parallel Depth-First Search, Parallel Best First Search. **CUDA Architecture:** CUDA Architecture, Using the CUDA Architecture, Applications of CUDA Introduction to CUDA C-Write and launch CUDA C kernels, Manage GPU memory, Manage communication and synchronization, Parallel programming in CUDA- C

Text Books:

1. Ananth Grama, Anshul Gupta, George Karypis, and Vipin Kumar, "Introduction to Parallel Computing", 2nd edition, Addison-Wesley, 2003, ISBN: 0-201-64865-2
2. Jason sanders, Edward Kandrot, "CUDA by Example", Addison-Wesley, ISBN-13: 978-0-13-138768-3

Reference Books:

1. Kai Hwang, "Scalable Parallel Computing", McGraw Hill 1998, ISBN:0070317984
2. Shane Cook, "CUDA Programming: A Developer's Guide to Parallel Computing with GPUs", Morgan Kaufmann Publishers Inc. San Francisco, CA, USA 2013 ISBN: 9780124159884
3. David Culler Jaswinder Pal Singh, "Parallel Computer Architecture: A Hardware/Software Approach", Morgan Kaufmann,1999, ISBN 978-1-55860-343- 1
4. Rod Stephens, "Essential Algorithms", Wiley, ISBN: 978-1-118-6



IV Year I Semester	OBJECT ORIENTED PROGRAMMING THROUGH JAVA-V231414451	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course are to

- Identify Java language components and how they work to get her in applications
- Learn the fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries.
- Learn how to extend Java classes with inheritance and dynamic binding and how to use exception handling in Java applications
- Understand how to design applications with threads in Java
- Understand how to use Java APIs for program development

COURSE OUTCOMES: At the end of the course, students will be able to

- CO1. Write Java programs using basic concepts, data types, and control statements.
- CO2. Design and implement classes, objects, and methods.
- CO3. Work with arrays, inheritance, and interfaces.
- CO4. Handle exceptions and use packages.
- CO5. Develop multithreaded and GUI applications using Java FX and JDBC.

UNIT I:

Object Oriented Programming: Basic concepts, Principles, Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

Data Types, Variables, and Operators : Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator (=), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators. Control Statements: Introduction, if Expression, Nested if Expressions, ifelse Expressions, Ternary Operator?;, Switch Statement, Iteration Statements, while Expression, do-while Loop, for Loop, Nested for Loop, For-Each for Loop, Break Statement, Continue Statement.

UNIT II:

Classes and Objects: Introduction, Class Declaration and Modifiers, Class Members, Declaration of Class Objects, Assigning One Object to Another, Access Control for Class Members, Accessing Private Members of Class, Constructor Methods for Class, Overloaded Constructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

Methods: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static.



UNIT III:

Arrays: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, Operations on Array Elements, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three dimensional Arrays, Arrays as Vectors.

Inheritance: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance. Interfaces: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

UNITIV:

Packages and Java Library: Introduction, Defining Package, Importing Packages and Classes in to Programs, Path and Class Path, Access Control, Packages in Java SE, Java. Lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto boxing and Auto-un boxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, Temporal Adjusters Class, Temporal Adjusters Class.

Exception Handling: Introduction, Hierarchy of Standard Exception Classes, Key words throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exceptions.

Java I/O and File: Java I/OAPI, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

UNIT V:

String Handling in Java: Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer.

Multithreaded Programming: Introduction, Need for Multiple Threads Multithreaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread Priority-Synchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

Text Books:

1. JAVA one step ahead, Anitha Seth, B.L. Juneja, Oxford.
2. Joy with JAVA, Fundamentals of Object-Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, 2023.
3. JAVA9 for Programmers, Paul Deitel, Harvey Deitel, 4thEdition, Pearson.

References Books:

1. The complete Reference Java, 11thedition, HerbertSchildt, TMH
2. Introduction to Java programming, 7thEdition, YDanielLiang, Pearson

Online Resources:

1. <https://nptel.ac.in/courses/106/105/106105191/>
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_012880464547618816



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

[347_shared/overview](#)



IV Year I Semester	COMPUTER NETWORKS-V231414452	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objective of this course is to

- To understand the different types of networks
- To develop an understanding the principles of computer networks.
- To familiarize with Reference model OSI and TCP/IP
- To understand various layers of Reference models functions
- To explore network protocols

COURSE OUTCOMES: After completion of the course, students will be able to

- CO1. Understand the reference models and network protocols
- CO2. Describe data transmission media and data link layer.
- CO3. Understand the network layer design issues and Network layer Protocols.
- CO4. Evaluate transport layer services and its protocols.
- CO5. Understand application layer protocols and their uses

UNIT - I: Introduction

Types of Computer Networks, Reference Models- The OSI Reference Model, The TCP/IP Reference Model, A Critique of the OSI Model and Protocols, A Critique of the TCP/IP Reference Model. History of Internet.

UNIT -II: The Data Link Layer

Transmission Media, Guided and Un-guided media, Data Link Layer Design Issues, Services Provided to the Network Layer, Error detecting and Error Correcting codes, Elementary Data Link Protocols, Sliding Window Protocols, HDLC, PPP. Multiple Access Protocols Wired Lans: Ethernet, Fast Ethernet, Gigabit Ethernet

UNIT - III: The Network Layer

Network Layer Design Issues, Routing Algorithms, Congestion, Congestion control algorithms. The Network Layer in the Internet, The IP Version 4 Protocol, IP Addresses Classful, CIDR, NAT, IP Version 6 Protocol, Transition from IPV4 to IPV6

UNIT -IV: The Transport Layer

The Transport Layer Services, Transport Layer Protocols: UDP, TCP and SCTP

UNIT -V: The Application Layer

The World Wide Web, HTTP, Domain Name Space, Remote Logging, Electronic Mail and File Transfer

Textbooks:

1. “Computer Networks”, Andrew S Tanenbaum, David J Wetherall, 5th Edition, Pearson
2. “Data Communications and Networking”, Behrouz A Forouzan, 4th Edition, Tata McGraw Hill Education



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

Reference Books:

2. "TCP/IP Protocol Suite", Behrouz Forouzan, McGraw Hill.
1. "Data and Computer Communication", William Stallings,
Pearson



IV Year I Semester	SOFTWARE ENGINEERING-V231414453	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES: The main objectives of this course is to

- Evolution of software engineering, software development projects, and life cycle models
- Understand software project management complexities, estimation techniques, and risk management
- Apply requirements analysis and specification, software design principles, and user interface design
- Understand coding, testing, debugging, and software reliability and quality management
- Introduce CASE tools, software maintenance, and software reuse concepts

COURSE OUTCOMES: After completion of the course, students will be able to

- CO1. Explain software engineering concepts and life cycle models.
- CO2. Apply software project management techniques and estimation methods.
- CO3. Analyze and design software systems using requirements analysis and specification.
- CO4. Test and maintain software systems using coding, testing, and debugging techniques.
- CO5. Apply CASE tools and software reuse concepts to software development projects.

UNIT - I:

Introduction: Evolution, Software development projects, Exploratory style of software developments, Emergence of software engineering, Notable changes in software development practices, Computer system engineering.

Software Life Cycle Models: Basic concepts, Waterfall model and its extensions, Rapid application development, Agile development model, Spiral model.

UNIT - II:

Software Project Management: Software project management complexities, Responsibilities of a software project manager, Metrics for project size estimation, Project estimation techniques, Empirical Estimation techniques, COCOMO, Halstead's software science, risk management.

Requirements Analysis and Specification: Requirements gathering and analysis, Software Requirements Specification (SRS), Formal system specification, Axiomatic specification, Algebraic specification, Executable specification and 4GL.

UNIT - III:

Software Design: Overview of the design process, How to characterize a good software design? Layered arrangement of modules, Cohesion and Coupling. approaches to software design.

Agility: Agility and the Cost of Change, Agile Process, Extreme Programming (XP), Other Agile Process Models, Tool Set for the Agile Process (Text Book 2)



Function-Oriented Software Design: Overview of SA/SD methodology, Structured analysis, Developing the DFD model of a system, Structured design, Detailed design, and Design Review.

User Interface Design: Characteristics of a good user interface, Basic concepts, Types of user interfaces, Fundamentals of component-based GUI development, and user interface design methodology.

UNIT - IV:

Coding And Testing: Coding, Code review, Software documentation, Testing, Black-box testing, White-Box testing, Debugging, Program analysis tools, Integration testing, testing object-oriented programs, Smoke testing, and some general issues associated with testing.

Software Reliability and Quality Management: Software reliability. Statistical testing, Software quality, Software quality management system, ISO9000. SEI Capability maturity model. Few other important quality standards and Six Sigma.

UNIT - V:

Computer-Aided Software Engineering (Case): CASE and its scope, CASE environment, CASE support in the software life cycle, other characteristics of CASE tools, Towards second generation CASE Tool, and Architecture of a CASE Environment.

Software Maintenance: Characteristics of software maintenance, Software reverse engineering, Software maintenance process models and Estimation of maintenance cost.

Software Reuse: Reuse-definition, Introduction, Reason behind no reuses of ar, Basic issues in any reuse program, A reuse approach, and Reuse at organization level.

Text Books:

1. Fundamentals of Software Engineering, RajibMall, 5th Edition, PHI.
2. Software Engineering a Practitioner's Approach, Roger S. Pressman, 9th Edition, McGraw Hill International Edition.

Reference Books:

1. Software Engineering, Ian Sommerville, 10th Edition, Pearson.
2. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.

Resources:

1. <https://nptel.ac.in/courses/106/105/106105182/>
2. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_01260589506387148827_shared/overview35_shared/overview
3. https://infyspringboard.onwingspan.com/web/en/app/toc/lex_auth_0133826904110039047



IV Year I Semester	IOT BASED SMART SYSTEMS-V231414454	L	T	P	C
		3	0	0	0

COURSE OBJECTIVES: The main objective of this course is to

- To study about Internet of Things technologies and its role in real time applications.
- To introduce the infrastructure required for IoT
- To familiarize the accessories and communication techniques for IoT.
- To provide insight about the embedded processor and sensors required for IoT
- To familiarize the different platforms and Attributes for IoT

COURSE OUTCOMES: At the end of this course, the students will have the ability to

CO1. Analyze the concepts of IoT and its present developments.

CO2. Compare and contrast different platforms and infrastructures available for IoT

CO3. Explain different protocols and communication technologies used in IoT

CO4. Analyze the big data analytic and programming of IoT

CO5. Implement IoT solutions for smart applications

UNIT I

INTRODUCTION TO INTERNET OF THINGS: Overview, Hardware and software requirements for IOT, Sensor and actuators, Technology drivers, Business drivers, Typical IoT applications, Trends and implications.

UNIT II

IOT ARCHITECTURE: IoT reference model and architecture -Node Structure –Sensing, Processing, Communication, Powering, Networking – Topologies, Layer/Stack architecture, IoT standards, Cloud computing for IoT, Bluetooth, Bluetooth Low Energy beacons.

UNIT III

PROTOCOLS AND WIRELESS TECHNOLOGIES FOR IOT PROTOCOLS: NFC, SCADA and RFID, Zigbee MIPI, M-PHY, UniPro, SPMI, SPI, M-PCIe GSM, CDMA, LTE, GPRS, small cell. Wireless technologies for IoT: Wi-Fi (IEEE 802.11), Bluetooth/Bluetooth Smart, ZigBee Smart, UWB (IEEE 802.15.4), 6LoWPAN, Proprietary systems-Recent trends.

UNIT IV

IOT PROCESSORS Services/Attributes: Big-Data Analytics for IOT, Dependability, Interoperability, Security, Maintainability.

Embedded processors for IOT: Introduction to Python programming -Building IOT with RASPBERRY PI and Arduino.

UNIT V

CASE STUDIES: Industrial IoT, Home Automation, smart cities, Smart Grid, connected vehicles, electric vehicle charging, Environment, Agriculture, Productivity Applications, IOT Defense



**SREE VAHINI INSTITUTE OF SCIENCE AND TECHNOLOGY, TIRUVURU
AUTONOMUS**

V23 B.Tech IV Year CSE (DS) COURSE STRUCTURE & SYLLABUS

REFERENCES:

1. Arshdeep Bahga and VijaiMadiseti: A Hands-on Approach “Internet of Things”, Universities Press 2015.
2. Vijay Madiseti , ArshdeepBahga, “Internet of Things (A Hands on-Approach)”, 2014.



IV Year I Semester	QUANTUM SCIENCE AND TECHNOLOGY V231414455	L	T	P	C
		3	0	0	3

COURSE OBJECTIVES:

- To introduce fundamental concepts of quantum mechanics and its mathematical formalism.
- To explore quantum computing and communication principles and technologies.
- To understand the physical implementation and limitations of quantum systems.
- To enable students to relate quantum theory to practical applications in computing, cryptography, and sensing.
- To familiarize students with the emerging trends in quantum technologies.

COURSE OUTCOMES: After completing this course, students will be able to

- CO1. Explain core principles of quantum mechanics and their technological implications.
CO2. Analyze quantum phenomena like superposition and entanglement.
CO3. Apply mathematical tools to model and solve quantum systems.
CO4. Demonstrate understanding of quantum algorithms and quantum circuits.
CO5. Evaluate potential applications and challenges in quantum communication and sensing.

Unit 1: Fundamentals of Quantum Mechanics: Historical background: Blackbody radiation, photoelectric effect, and Compton scattering; Dual nature of light and matter; De Broglie hypothesis; Schrödinger equation; Free particle, infinite potential well, step potential; Operators and observables: position, momentum, Hamiltonian; Commutation relations and uncertainty principle; Quantum postulates and measurement theory; Eigen values, eigenfunctions.

Unit 2: Quantum Information Theory: Classical vs. quantum information; Qubit representation using Bloch sphere; Quantum superposition and quantum entanglement; Dirac notation (bra-ket), tensor products, and composite systems; Bell states and EPR paradox; Quantum gates: Pauli-X, Y, Z; Hadamard; Phase; T; CNOT; Quantum circuit models and notation; Measurement in computational basis; Quantum teleportation and no-cloning theorem; Quantum state tomography (introductory)

Unit 3: Quantum Computing: Classical computing review and limitations; Quantum parallelism and Interference; Deutsch and Deutsch-Jozsa algorithms; Grover's search algorithm, Oracle and amplitude amplification; Shor's factoring algorithm (overview and significance); Quantum Fourier Transform (QFT); Quantum error correction: Bit-flip, phase-flip, and Shor's 9-qubit code; Introduction to quantum programming: Qiskit, Cirq, IBM Quantum Experience (overview)

Unit 4: Quantum Communication: Introduction to quantum cryptography; Quantum key distribution (QKD): BB84 protocol; Entanglement-based QKD: Ekert protocol (E91); Eaves dropping and security of QKD; Quantum teleportation (circuit and protocol); Quantum dense coding; Quantum networks and entanglement swapping; Role of quantum repeaters; Single-photon sources and detectors; Implementation challenges (loss, decoherence, noise)



Unit 5: Quantum Technologies and Applications: Quantum sensors: magnetometry, gravimetry; Quantum metrology: standard time, atomic clocks; Quantum imaging and lithography; Quantum materials: topological insulators, graphene, quantum dots; NV centers in diamonds for sensing; Hardware platforms: Superconducting qubits, Trapped ions, Photonic quantum processors; Quantum supremacy and NISQ era; Global initiatives: IBM, Google, D-Wave, IonQ, India's NQM; Ethical concerns and future prospects

Text Books:

1. "Quantum Computation and Quantum Information" by Michael A. Nielsen and Isaac L. Chuang
2. "Quantum Mechanics: Concepts and Applications" by Nouredine Zettili

Online Learning Resources:

1. <https://nptel.ac.in/courses/104104082>
2. <https://nptel.ac.in/courses/115104096>
3. <https://nptel.ac.in/courses/122106034>



IV Year I Semester	FULL STACK DEVELOPMENT-2 V231414461 (Skill Enhancement Course)	L	T	P	C
		0	1	2	2

COURSE OBJECTIVES: The main objectives of the course are to

- Make use of router, template engine and authentication using sessions to develop application in Express JS.
- Build a single page application using Restful APIs in Express JS
- Apply router and hooks in designing ReactJS application
- Make use of MongoDB queries to perform CRUD operations on document database

COURSE OUTCOMES: On completion of the course, student will be able to

- CO1. Develop web applications using Express JS: Design and implement web applications using Express JS, Including routing, middleware, templating, and form data handling.
- CO2. Implement authentication and session management: Use cookies, sessions, and authentication mechanisms to secure web applications.
- CO3. Interact with databases using MongoDB and Mongoose: Connect to MongoDB databases using mongoose and perform CRUD operations.
- CO4. Build RESTful APIs: Design and implement RESTful APIs using ExpressJS and MongoDB.
- CO5. Develop front-end applications using ReactJS: Build front-end applications using ReactJS, including rendering HTML, JSX, components, props, states, and event handling .

Experiments covering the Topics:

- Express JS – Routing, HTTP Methods, Middleware, Templating, Form Data
- Express JS – Cookies, Sessions, Authentication, Database, RESTful APIs
- ReactJS – Render HTML, JSX, Components – function & Class, Props and States, Styles, Respond to Events
- ReactJS – Conditional Rendering, Rendering Lists, React Forms, React Router, Updating the Screen
- ReactJS – Hooks, Sharing data between Components, Applications – To-do list and Quiz
- MongoDB – Installation, Configuration, CRUD operations, Databases, Collections and Records

Sample Experiments:

1. Express JS – Routing, HTTP Methods, Middleware.

- a. Write a program to define a route, Handling Routes, Route Parameters, Query Parameters and URL building.
- b. Write a program to accept data, retrieve data and delete a specified resource using http methods.



- c. Write a program to show the working of middleware.

2. Express JS – Templating, Form Data

- a. Write a program using templating engine.
- b. Write a program to work with form data.

3. Express JS – Cookies, Sessions, Authentication

- a. Write a program for session management using cookies and sessions.
- b. Write a program for user authentication.

4. Express JS – Database, RESTful APIs

- a. Write a program to connect MongoDB database using Mongoose and perform CRUD operations.
- b. Write a program to develop a single page application using RESTful APIs.

5.ReactJS – Render HTML, JSX, Components – function & Class

- a. Write a program to render HTML to a web page.
- b. Write a program for writing markup with JSX.
- c. Write a program for creating and nesting components (function and class).

d.ReactJS – Props and States, Styles, Respond to Events

- a. Write a program to work with props and states.
- b. Write a program to add styles (CSS & Sass Styling) and display data.
- c. Write a program for responding to events.

e. ReactJS – Conditional Rendering, Rendering Lists, React Forms

- a. Write a program for conditional rendering.
- b. Write a program for rendering lists.
- c. Write a program for working with different form fields using react forms.

f. ReactJS – React Router, Updating the Screen

- a. Write a program for routing to different pages using react router.
- b. Write a program for updating the screen.

g. ReactJS – Hooks, Sharing data between Components

- a. Write a program to understand the importance of using hooks.
- b. Write a program for sharing data between components.

h. MongoDB – Installation, Configuration, CRUD operations

- a. Install MongoDB and configure ATLAS
- b. Write MongoDB queries to perform CRUD operations on document using insert(), find(), update(), remove()



i. MongoDB – Databases, Collections and Records

- a. Write MongoDB queries to Create and drop databases and collections.
- b. Write MongoDB queries to work with records using find(), limit(), sort(), create Index(), aggregate().

j. Augmented Programs: (Any 2 must be completed)

- a. Design a to-do list application using NodeJS and Express JS.
- b. Design a Quiz app using ReactJS.
- c. Complete the MongoDB certification from MongoDB University website.

Text Books:

1. Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node, Vasanth Subramanian, 2nd edition, APress, O'Reilly.
2. Node.js in Action, Mike Cantelon, Mark Harter, T.J. Holowaychuk, Nathan Rajlich, Manning Publications. (Chapters 1-11)
3. React Quickly, Azat Mardan, Manning Publications (Chapters 1-8, 12-14)

Web Links:

1. ExpressJS - <https://www.tutorialspoint.com/expressjs>
 2. ReactJS - <https://www.w3schools.com/REACT> (and) <https://react.dev/learn#>
- MongoDB - <https://learn.mongodb.com/learning-paths/introduction-to-mongodb>



IV Year I Semester	CONSTITUTION OF INDIA-V2314144C1 (Audit Course)	L	T	P	C
		2	0	0	--

COURSE OBJECTIVES: The main objectives of the course are to

- Understand the premises informing the twin themes of liberty and freedom from a civil rights perspective.
- To address the growth of Indian opinion regarding modern Indian intellectuals' constitutional role and entitlement to civil and economic rights as well as the emergence of nationhood in the early years of Indian nationalism.
- To address the role of socialism in India after the commencement of the Bolshevik Revolution in 1917 and its impact on the initial drafting of the Indian Constitution.

COURSE OBJECTIVES: On completion of the course, student will be able to

CO1. Explain the salient features of the Indian Constitution.

CO2. Analyze the relationship between fundamental rights and directive principles of state policy.

CO3. Describe the powers and functions of the president, governor, and council of ministers.

CO4. Understand the importance of local governance and the role of elected representatives.

CO5. Students will be able to appreciate the role of the election commission in ensuring free and Fair elections.

UNIT-I: History of Making of the Indian Constitution: History, Drafting Committee, (Composition & Working) Philosophy of the Indian Constitution- Preamble, Salient, Features

UNIT-II: Contours of Constitutional Rights & Duties: Fundamental Rights, Right to Equality, Right to Freedom, Right against Exploitation, Right to Freedom of Religion, Cultural and Educational Rights, Right to Constitutional Remedies, Directive Principles of State Policy, Fundamental Duties.

UNIT-III: Organs of Governance: Parliament, Composition, Qualifications and Disqualifications, Powers and Functions, Executive- President, Governor, Council of Ministers, Judiciary, Appointment and Transfer of Judges, Qualifications, Powers and Functions

UNIT-IV: Local Administration: District's Administration head: Role and Importance, Municipalities: Introduction, Mayor and role of Elected Representative CEO of Municipal Corporation, Pachayati raj: Introduction, PRI: Zila Pachayat, Elected officials and their roles, CEO Zila Pachayat: Position and role, Block level: Organizational Hierarchy (Different departments),



Village level: Role of Elected and Appointed officials, Importance of grass root democracy

UNIT-V: Election Commission: Election Commission: Role and Functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and Functioning, Institute and Bodies for the welfare of SC/ST/OBC and women.

Text Books:

1. The Constitution of India, 1st Edition, (Bare Act), Government Publication, 1950
2. Framing of Indian Constitution, 1st Edition, Dr. S. N. Busi, Dr. B. R. Ambedkar 2015

Reference Books:

1. Indian Constitution Law, 7th Edition, M. P. Jain, Lexis Nexis, 2014.